CLAIMS

What is claimed is:

1. A method for optimizing the processing instructions through a processor, comprising the steps of:

(A) processing first like instructions through two or more instructions paths of the processor, each of the paths having different heuristics associated therewith;

(B) monitoring progress of the first like instructions through the instruction paths;

(C) determining which of the instruction paths is a first leader in processing the first like instructions; and

(D) modifying heuristics of one or more of the instruction paths based on heuristics of the first leader.

2. A method of claim 1, further comprising grouping the first like instructions as a bundle from a common program thread.

3. A method of claim 1, the step of modifying comprising modifying heuristics of each of the instruction paths.

4. A method of claim 1, the step of modifying comprising modifying heuristics of instruction paths other than heuristics of the leader.

5. A method of claim 1, further comprising processing the first like instructions through the leader without being affected by the step of modifying.

6. A method of claim 1, further comprising processing additional instructions from a program thread of the first like instructions through the multiple instruction paths and without redundancy.

7. A method of claim 1, further comprising:

processing second like instructions through two or more instructions paths of the processor, each of the paths having different heuristics associated therewith;

monitoring progress of the second like instructions through the instruction paths;

determining which of the instruction paths is a second leader in processing the instructions; and

modifying heuristics of one or more of the instruction paths based on heuristics of the second leader.

8. A method of claim 1, the step of modifying comprising modifying one or more of CPU-bound heuristics and memory-bound heuristics.

9. A method of claim 1, the step of modifying comprising modifying heuristics based upon one or more of branch prediction and prefetch heurisitcs.

10. A method of claim 1, further comprising repeating steps (A)-(D) for modified heuristics within multiple instruction paths to asymptotically approach optimized characteristics for the instruction paths.

11. A processor for processing program instructions, comprising:

at least two parallel instruction paths, each of the paths having an array of pipeline execution units and associated heuristics affecting how the instructions are processed therein; and

assessment logic for monitoring processing of the instructions within the paths and for modifying the heuristics of at least one of the paths to improve per thread performance of the processor.

12. A system of claim 11, the heuristics of each of the instruction paths having one or more of fetch heuristics, execution heuristics, and cache heuristics.

13. A system of claim 11, the two parallel instruction paths comprising parallel core processors on a common die.

14. A system of claim 11, the parallel instruction paths constructed and arranged to initially process first like instructions therethrough, the assessment logic monitoring the processing of the first like instructions to determine optimized heuristics for the instruction paths.

15. A system of claim 14, the parallel instruction paths constructed and arranged to subsequently process different instructions therethrough to improve per thread processing performance.

16. A system of claim 14, the parallel instruction paths constructed and arranged to subsequently process second like instructions therethrough, the assessment logic monitoring the processing of the second like instructions to determine optimized heuristics for the instruction paths.

17. A system of claim 11, each of the parallel instruction paths forming a cluster constructed and arranged to process instructions as bundles.

18. A system of claim 11, the parallel instruction paths and assessment logic cooperating to process one or more bundles of like instructions through the instruction paths to monitor and then modify heuristics of the instruction paths to improve per thread processing of the instructions.

19. In a processor of the type having at least two parallel instruction paths, each of the paths having an array of pipeline execution units and associated heuristics affecting how the instructions are processed, the improvement comprising:

assessment logic for monitoring processing of the instructions within the paths and for modifying the heuristics of at least one of the paths to improve per thread performance of the processor.

20. In a processor of claim 19, the further improvement wherein the parallel instruction paths are constructed and arranged to initially process first like instructions therethrough, the assessment logic monitoring the processing of the first like instructions to determine optimized heuristics for the instruction paths.